Financial Management Service

Secure Payment System

SPS FPA CLIENT OVERVIEW

Version 1.01: September 3, 2002

REVISION HISTORY

Version	Date	Authors	Notes
V 1.0	08/01/02	WW	Initial Version
V 1.01	09/03/02	DB	Included DataKey Card Reader Software specifications (page 4)

INTRODUCTION

This document provides an overview of the Secure Payment System (SPS) FPA Client workstation environment. An elaboration of the software and hardware client components is detailed. The underlying intent is to discuss processing differences between Electronic Certification System (ECS) and SPS with the introduction of internet technology and Public Key Infrastructure (PKI) security services. These processing needs affect usability concerns, and are influenced by connectivity options, and workstation processor selection.

TARGET AUDIENCE

1) Current users of ECS
2) Federal Program Agency network security support personnel
3) Desktop Configuration Management personnel

BACKGROUND

The Department of the Treasury/Financial Management Service (FMS) is developing a replacement for its Electronic Certification System (ECS). Federal Program Agencies use ECS to prepare and submit authorizations to FMS to issue payments. It also allows Regional Finance Center's a means of extracting approved payment schedules for payment execution (e.g. check printing, electronic funds transfer).

ECS has been in operation since 1987 and uses symmetric cryptographic techniques to provide proof of individual identity and assurance that each payment request has not been modified. The ECS also employs a key management system to generate, distribute, and delete cryptographic keys. The cryptographic components supporting the ECS need to be updated with current Public Key Infrastructure (PKI) technologies to ensure adequate security is maintained.

ECS must be run from a dedicated PC at the FPA. ECS is a typical "fat client" application. A "fat client" is one in which most of the application resides on and runs on the user workstation. ECS has built its own user interface and its payment data is created in local data files, prior to transmission. ECS is run from a strictly DOS environment on legacy PCs (486s).

SPS will use more current hardware and software technologies. SPS will introduce internet client/server and web browser technologies for payment schedule creation and transmission. Cryptography of payment schedules will be supported using PKI asymmetric ciphers versus ECS' traditional symmetric ciphers. Java applets will run on the client providing XML digital signing techniques. Though primarily WINTEL, SPS can be run from any PC meeting the minimum hardware and software requirements.

SPS will allow personnel at agency locations to submit schedules to FMS over a browser/web interface connecting to an FMS web site. Dial-up to FMS, with FMS acting as the FPA ISP, will be available as a contingency, in the event of FPA network internet unavailability. Dial-up FMS ISP access will also be available for those agencies which

have policies that prohibit conducting official business over the internet or which may have restrictions regarding the downloading and processing of Java Mobile Code on their network clients.

FMS plans to begin rolling out the Secure Payment System (SPS) to customer FPAs during the first quarter of Calendar Year 2003.

MINIMUM SPS HARDWARE REQUIREMENTS

For the minimum requirements for SPS functionality, please note FMS recommends the USB Smart Card Reader for improved throughput. Since Windows 2000 and Windows XP operating systems offer USB support they are also recommended. A serial connection has a maximum throughput of 230 kilobits per second (kbps), while the Universal Serial Bus (USB) has a maximum possible throughput of 12Mbps (or 1.5MBps). Pentium III 500mhz

- 128mb (or higher) RAM
- 500mb (or more) free hard disk space
- 2x (or faster) CDROM drive
- One free Serial or USB port (for smart card reader)
- One 56K external or internal modem for dial-up connectivity, with analog line

  And or
- 10mbps (or faster) network interface card
- DataKey 330D Smart Card
- DataKey Card Reader

  DKR610 Serial Port Reader

  or

  DKR630 USB Reader

SPS CLIENT SOFTWARE

- Operating Systems (one of the following)

  Windows 98 Second Edition

  NT 4.0 SP6

  Windows 2000 SP2

  Windows XP
- DataKey Card Reader Software

  DataKey CIP Software Version 4.6.1

- Browser

  Internet Explorer 5.5 or higher

  or

  Netscape 4.76

- Java Plug-in to Browser

  Java Plug-in 1.3.1_04 (Sun, http://java.sun.com/j2se/1.3/)

Java plug-in software enables enterprise customers to direct java applets on their web pages to run using Sun's Java 2 Runtime Environment, Standard Edition (JRE), instead of the web browser's default virtual machine. This provides a common framework for running desktop client Java applets across multiple browsers, and browser versions. The use of the Java Plug-in JRE or the browser's internal JRE is configurable either within the browser's settings, or by use of the Java Plug-in Control Panel.

PROCESSING DIFFERENCES ECS AND SPS

Although SPS can be considered to fit the thin client paradigm, client resources are critical in the client signature (Certifying Officer or Data Entry Operator) process. Thin client refers to client programs that display in a browser, but execution of user code primarily takes place on a central web server, not the desktop PC. The exception to this by SPS, is the digital signature process when signing schedules with large amounts of data at one time. When signing data using the smart card, a message digest is computed on the workstation client using SHA-1 (Secure Hashing Algorithm). Actual digital signatures are generated or verified on the smart card using the message digests computed on the workstation. The length of time required to generate a digital signature is a function of the payment schedule size. The relative key strength has been increased from ECS to SPS. A workstation with a faster processor may provide more desirable results.

ECS processes are segmented tasks. A single task would require certification of the schedule. This process creates a message digest (checksum, or Message Authentication Code) of the plain text data, and then encrypts the digest. The next segmented task would require transmission of the certified schedule, both plain text data plus the encrypted message digest to the mainframe. In SPS, the following can be viewed to a typical end user as comprising one task: (this sequence is simplified, for illustrative purposes)

1) Browser request for schedule
2) Web server response, downloads schedule to client browser
3) Client signature of data displayed by browser
4) Posting of signature and data to web server (upload)
5) Web server signature and response to browser of confirmation of successful transaction

In SPS not only is the message digest encrypted. Using Secure Sockets Layer (SSL) the entire contents of the XML schedule are encrypted, including XML descriptive identifiers. All this is accomplished when a CO selects a schedule to sign (certify). Thus while the ECS certify task might appear to require less user waiting time than the SPS signing activity, SPS is combining several activities into one transaction. Also, the amount of data (number of bytes) transferred (both downloaded and uploaded) is greater in SPS than in ECS. As the discussion illustrates, high file transfer speeds may also be critical for rapid execution of signature transactions.

SPS FPA CLIENT

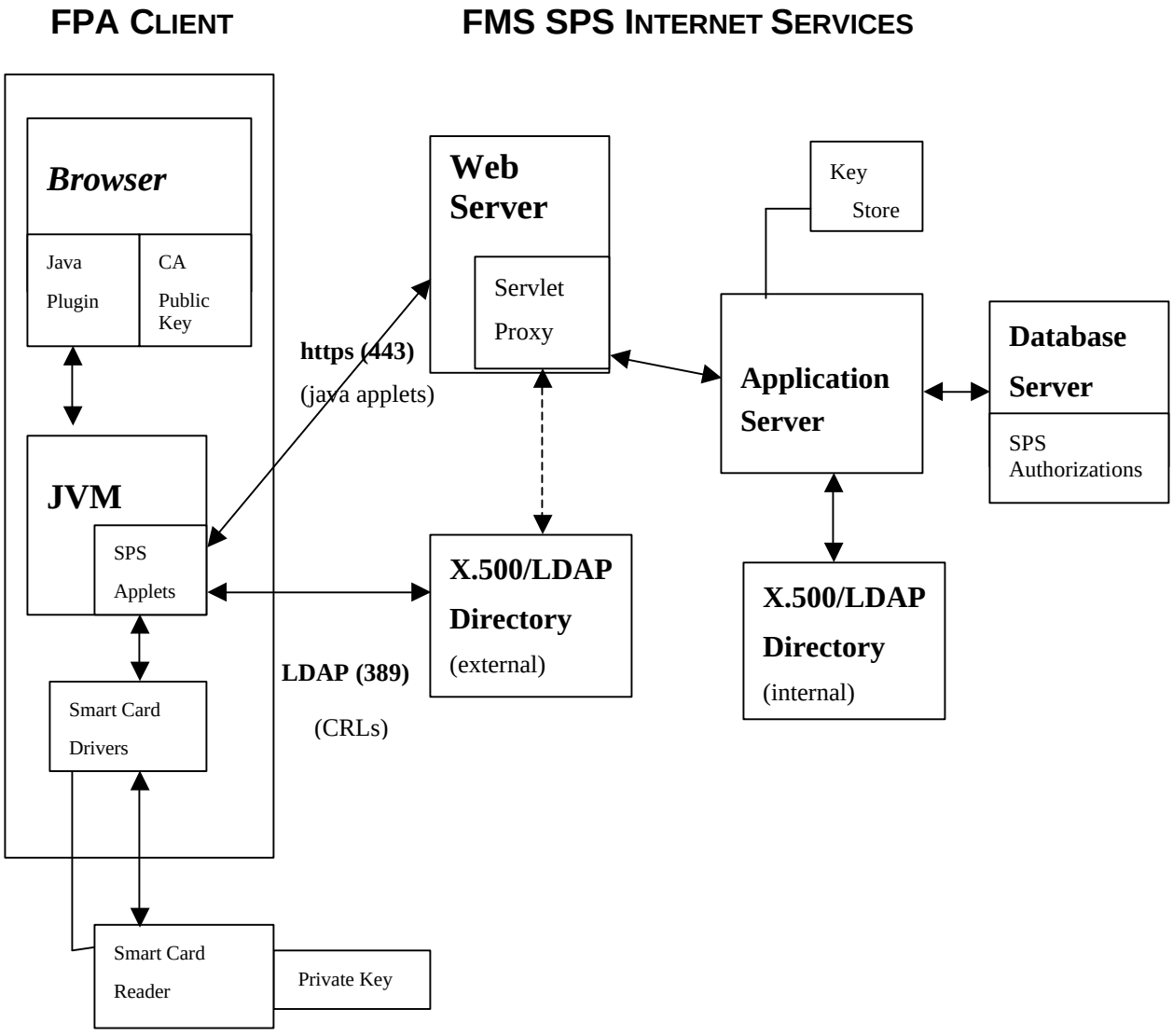


Figure 1 - Business View of SPS Client Processes

This diagram portrays a broad picture of traffic between a typical FPA SPS client and FMS Internet Services. It's provided for those FPA's considering using networked higher speed internet connections. Only HTTP, SSL, protected SSL LDAP, and DNS will travel between the FPA SPS user client region and the external FMS web services.

SPS will utilize directories in order to access and verify certificates. The SPS client will use a protected SSL LDAP to access external directories located in a network that is accessible to the users. The application will use an internal directory on its secured internal network. All of the directories will be replicated from the master directory that is populated by the CA.

Connectivity Options

- **Existing LAN connectivity to FMS**

  Agencies already connected to FMS for other applications, could arrange for connectivity to the SPS application firewall via existing links, routers, and firewalls. Provides potential bandwidth greater than dial-up.

- **Internet access via LAN**

  Agencies having Internet access through their enterprise LAN, with or without a web proxy, can use SPS directly. Agencies must allow LDAP, PKI registration port access, and mobile Java code through their firewalls. Provides potential bandwidth greater than dial-up.

- **Internet access via ISP**

  Any ISP method that provides for standard web access can be used by the SPS application. This includes DSL, ISDN w/ISP, cable modem, dial-up of most kinds, or any other standard method. Standard OS features would be used to complete this connection. (Dial-up, would require phone number, userid, and password.)

- **Direct Dial-up to FMS**

  This option involves FMS having dial-in terminal servers outside the SPS application firewall. This is equivalent in most ways to dialing an ISP. Standard OS features would be used to complete this connection. This connect mode is intended to be backup if other more preferable modes fail.

- **VPN (Virtual Private Network)**

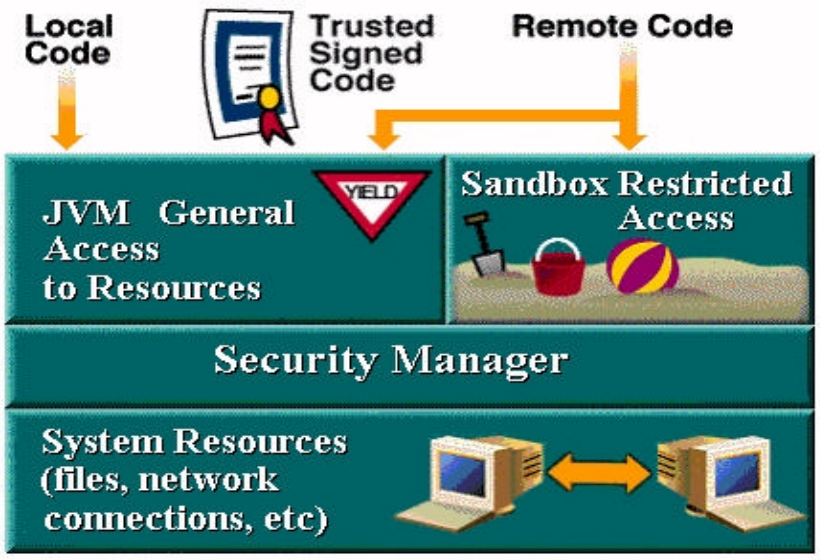
  A software or hardware VPN system (SSH, IPSec via L2TP, or PPTP via L2TP). Possibly used for SPS PKI registration purposes.

The hardware and connectivity options are less restrictive in SPS than in ECS. The selection of processor and connectivity speeds will be influenced by FPA workload and also FPA security restrictions. The amount of data (number of payments) in a schedule

affects the time in which a schedule is downloaded from the web server to the client browser. The amount of time necessary to sign a schedule is also a function of size. When data is signed and updated there also is a posting (uploading) time from browser to server. The increase in bandwidth may provide a measured impact on the speed of processing a transaction incorporating significant amounts of data. However, FPA security policy, (not opening LDAP ports, or blocking Java applets) may place restrictions on connectivity modes supported by their clients.

SPS AND JAVA MOBILE CODE

SPS requires XML payment schedule files (messages) to be signed on the client. This means SPS will be running outside what is commonly known as the Java "sandbox" to execute critical client processes. SPS will be using a "signed applet," and code signing certificates, as illustrated in the next figure. A digitally signed applet is treated like local code, with general access to resources, if the public key used to verify the signature is trusted. Unsigned applets are still run in the sandbox. Signed applets are delivered, with their respective signatures, in signed JAR (Java ARchive) files. SPS will present a code-signing certificate to the client browser for acceptance in downloading, installation and execution of the SPS Java applets. For those users familiar with IBM's Host on Demand secure mainframe access, the process of accepting, loading, and executing java applets within the desktop client is identical.



CONCLUSION

Each FPA must examine its own workload and make informed decisions on the best hardware/connectivity options that may satisfy their end-user experience. Modernization of ECS brings new solutions, but also a new set of security and usability concerns that may be addressed by more powerful hardware, increased bandwidth, or workload restructuring.